6LoWPAN Gateway System for Wireless Sensor Networks and Performance Analysis

Gopinath Rao Sinniah, Zeldi Suryady,
Usman Sarwar, Mazlan Abbas
Wireless Communication Cluster, MIMOS Berhad
Kuala Lumpur, Malaysia
{gopinath.rao, zeldi.suryady,
usman.sarwar, mazlan.abbas}@mimos.my

Sureswaran Ramadass
National Advanced IPv6 Centre of Excellence (NAv6)
Universiti Sains Malaysia (USM)
Pulau Pinang, Malaysia
sures@nav6.usm.my

Abstract*—The importance of Wireless Sensor Network to be connected to the Internet can be observed with the emergence of Internet of Things. Applications that require WSN nodes to be connected to the Internet has been steadily increasing over the years. Knowing the fact that these low capability devices cannot handle TCP/IP protocol stack, a new format has been introduced. IPv6 over Low Power Personal Area Network (6LoWPAN) enables these devices to be connected to the Internet seamlessly and the important network device that interconnects the WSN network and the Internet is the gateway. In this paper, a gateway system that manages the packets from both the WSN and the Internet is proposed. The system ensures that WSN nodes would be IP addressable and provides end-to-end connectivity. Two types of experiments to measure the functionalities, which are to provide end-to-end connectivity and performance on latency and transmission success rate are measured. A new packet format is also proposed with the elimination of the length field from the compressed UDP header. The experiment results showed that end-to-end communication was successfully established by allocating IPv6 address to the node at the gateway. Packet transmission success rate is 100% for 1 hop scenario while latency ranges from 60 and 145 ms and it is comparable with existing prior arts that ranges from 70 ms to few minutes.

***Index Terms*—6LoWPAN; Wireless Sensor Network; Gateway; IPv6; IEEE802.15.4.**

I. Introduction

This paper is an extension of work originally reported in The Sixth International Conference on Sensor Technologies and Applications (SENSORCOMM 2012) [1].

Wireless Sensor Network (WSN) has been increasingly being used since its introduction by DARPA in 1978. Usage of WSN gained momentum starting from early 2000 and with the cost reduced and better technology in place, more of these devices are being shipped. This is even more prevalent with the implementation of Internet of Things (IoT). Due to its hardware profile, WSN was only used in private and static network without any connectivity with other external devices. This has changed tremendously over the years. From a static type of connectivity to connectivity using web server and mobile network and now using TCP/IP protocol stack. The push for these technology is because the need and the benefits that it provides in various aspect of IoT ecosystems.

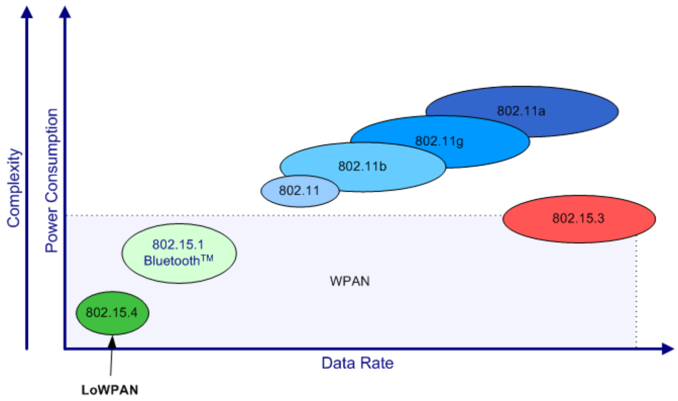


Fig. 1. Comparisons of IEEE802.15.4 with other Wireless Technologies

WSN nodes operate on low power, low processing and low memory hardware profile, which was defined in IEEE802.15.4 [2]. It is the same family of IEEE802.15 that specifies Wireless Personal Area Network (WPAN). Other standards in this family are Bluetooth (IEEE802.15.1) and High Rate WPAN (IEEE802.15.3). IEEE802.15.4 is also referred as Low Rate WPAN and has few revisions. The latest revision being standardized is IEEE802.15.4e and changes proposed in this revision are better channel hopping, which significantly increases robustness against external interference and persistent multi-path fading. IEEE 802.15.4 was designed to operate in three different bands as follows:

- 868.0 to 868.6 MHz → 1 Channel (data rates of 20 kbps, 100 kbps and 250 kbps)
- 902.0 to 928.0 MHz → 10 Channels (data rates of 40 kbps, 250 kbps)
- 2.40 to 2.48 GHz → 16 Channels (data rates of 250 kbps)

Even though there are three sets of bands for IEEE802.15.4, most of WSN implementations operate using 2.4 GHz frequency, which also being used by other standards such as WiFi and WiMAX and this leads to interference. Proper management of packet is required in WSN to reduce packet loss because of this interference. Figure 1 shows the comparison of IEEE802.15.4 standard with other wireless technologies in terms of complexities and power consumptions against data

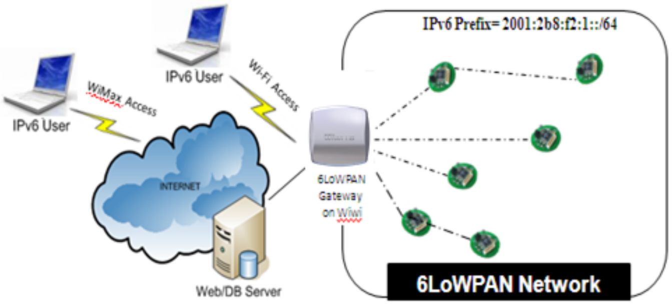


Fig. 2. Interconnection between WSN nodes and external network

rate.

Knowing the fact that existing TCP/IP is too bulky to be used in WSN nodes, 6LoWPAN [3] working grouping was created to provide a solution. The Working Group (WG) stated that the solution would be “pay as you use” header compression method that removes redundant or unnecessary network level information in the header. Some of the information can be derived from link-level IEEE802.15.4 header. Hence the 40 bytes IPv6 header was reduced to 2 bytes. This is achieved by reusing the link layer header information. The reduction of the header size is necessary as the total header size of IEEE802.15.4 is only 127 bytes, which is too small to accommodate the entire 40 bytes of IPv6 header.

There has been many solutions proposed to use 6LoWPAN to enable end-to-end communications between WSN nodes and external devices. All the communication from WSN nodes have to be through the gateway that interfaces between the WSN and external network. To enable the support for 6LoWPAN type of communication, a new system has to be developed on the gateway so that WSN nodes can be reachable in Internet and at the same time provides better performances. The gateway must be able to read all the three types of addressing format available in 6LoWPAN and also support other features such as routing, mobility, security and others. This paper extends the work provided in [1] and add contributions to [4], by providing detail gateway systems and performance analysis. The communication between the 6LoWPAN nodes and the external network through a gateway is given in Figure 2.

The main contribution of this paper are as follows:

a. Providing a detail 6LoWPAN gateway system that provides end-to-end communication between low power embedded wireless devices and external IPv6 devices.
b. A data management system on the gateway to handle packets that arrives both from external network and from Wireless Sensor Network, which results in increase of successful transmission of packets from WSN nodes and reduction in latency.

The rest of this article is organized as follows: Section II presents a new gateway system to handle communication from 6LoWPAN nodes. Section III provides the implementations and experiments to evaluate the performances, while Section IV discusses the results obtained. Section V presents the existing solutions related to WSN and specifically using 6LoWPAN. This paper concludes in Section VI with some suggestions for further research.

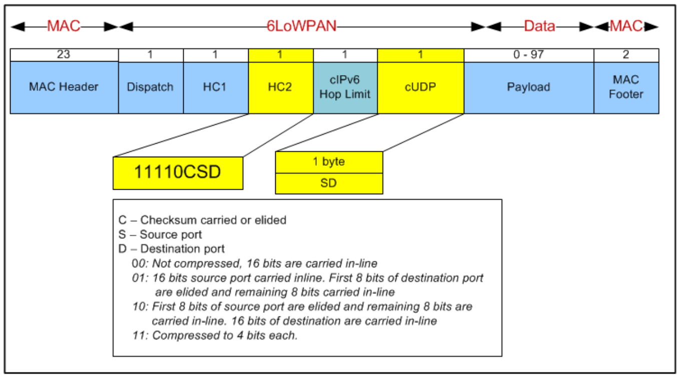

MAC
6LoWPAN
Data
MAC
23
1
1
1
1
1
0 - 97
2
MAC Header
Dispatch
HC1
HC2
cIPv6 Hop Limit
cUDP
Payload
MAC Footer
11110CSD
1 byte
SD
C – Checksum carried or elided
S – Source port
D – Destination port
00: Not compressed, 16 bits are carried in-line
01: 16 bits source port carried inline. First 8 bits of destination port are elided and remaining 8 bits carried in-line
10: First 8 bits of source port are elided and remaining 8 bits are carried in-line. 16 bits of destination are carried in-line
11: Compressed to 4 bits each.

Fig. 3. Changes to 6LoWPAN header

II. 6LoWPAN Gateway System

Architecture for extending the WSN to the Internet is presented by outlining the gateway that interfaces between the WSN and the Internet. By assigning IPv6 addresses and with proper handling of the packets, WSN nodes able to extend their reachability to the Internet and also supports two-way communications. The designed gateway system supports all the three addressing mechanisms available in the 6LoWPAN stack. The three addressing schemes are the short address (16 bits), MAC address (64 bits) and IPv6 address (128 bits). However, in our proposed solution, only 64-bit MAC address is used. This is because the 128-bit IPv6 address is too large to be used in IEEE802.15.4 header and 16-bit short address is not unique for WSN nodes identification. In this paper, only UDP type of packets are considered for experiments. Since the original 6LoWPAN header is not changed, the non-UDP packets will be treated as defined in the standards [20].

A. 6LoWPAN Header

Header Compression 2 (HC2) [19] [20], is a one byte field to define if UDP header need to be compressed or not. Bits 0 through 4 represent the next header ID and ’11110’ indicates the specific UDP header compression encoding. The 5th bit represent if checksum required or to be elided. Last 2 bits are used to define the source and destination ports. The header format is given in Figure 3. 16 bits of each used for source and destination port can be reduced to 4 bits each by eliminating the first 12 bits. With this, the compressed UDP header is only 1 byte, which is for the 4 bits source and destination ports each.

The 6LoWPAN header format used is given in Figure 4.

B. Extending WSN to Internet

The gateway is designed to support two standards of communication:

- **Pull based communication method** - IPv6 clients request data from sensor nodes in 6LoWPAN network. This

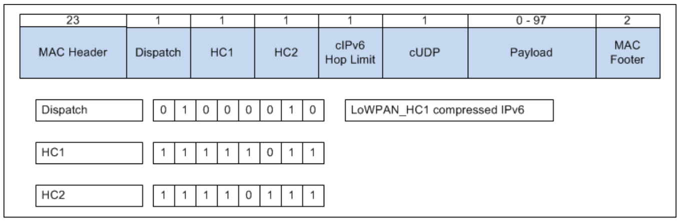


Fig. 4. 6LoWPAN Header Format used

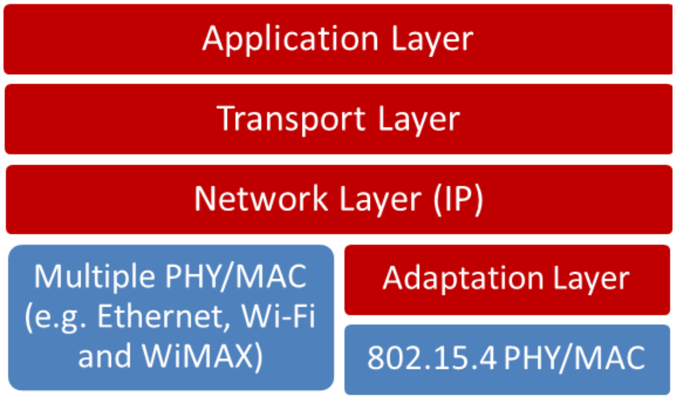


Fig. 5. Dual Stack Protocol in 6LoWPAN Gateway

is two-way communication, between client and sensor node.

- **Push based communication method** - Sensor nodes periodically send sensed data to a particular IPv6 client in IPv6 Network. The IPv6 client in this system is normally just like a remote station or database server. This is one-way communication, from sensor node to a remote station.

The 6LoWPAN gateway system aims at providing communication system and mechanism for ubiquitous wireless sensor network. The system is build by combining IEEE802.15.4 connectivity with standard interface to the Internet such as Wi-Fi, WiMAX, and Ethernet. The gateway must have dual stack protocol as shown in Figure 5 that represents multiple PHY/MAC (e.g., Ethernet, Wi-Fi, and WiMAX) for connecting external IP network and PHY/MAC of 6LoWPAN (IEEE 802.15.4).

Using the dual stack protocol, the gateway is designed to have 3 modules, which are:

- **6LoWPAN (WSN) Module** - This module consists of IEEE802.15.4 compliance hardware, which has the 6LoWPAN stack on it. The module is responsible for handling connectivity and data transmission of 6LoWPAN network using IEEE802.15.4 standard. Packets send by the sensor nodes are captured by this module and forwarded to the service module for further processing. It also forwards packets received from the service module to the sensor nodes.
- **External Interface Module** - This module defines the Physical and MAC layer of any interface that provides connectivity to external IP network. Therefore, the role of this module is to offer functionalities required to ensure connectivity to external IP network. Some of the interfaces may provide connectivity to LAN/Wireless LAN (e.g., Wi-Fi), while others can provide connectivity to back-haul internet (e.g., Ethernet or WiMAX). In case of gateway having multiple accesses, the selection of the interface depends on the priority configured in the service module and it could be changed manually.
- **Service Module** - This module provide services to handle both 6LoWPAN and IPv6 packets. It is a significant module that bridges all the interfaces that connects to different networks. Since most of the main processes occur in this module, the service module has a very important responsibility, which is integrating the 6LoWPAN network with the IP network through other external interfaces. The main purpose of this module is to provide functionalities for handling standard IPv6 packet from external network as well as 6LoWPAN packet. Two sub-modules are created to make this happen. The first sub-module is the node management that collects and stores all the necessary information of the sensor nodes. Some of the information stored are the MAC address of the sensor node, correspondence IPv6 address, and others. The second sub-module is for packet handling and translation. It handles both 6LoWPAN packets and IPv6 packets. The two types of transmission, which are pull-based and push-based are identified by the port number the packets are transmitted. The two sub-modules capture any IPv6 packet as well as 6LoWPAN packet, analyse the source and destination address and process accordingly.

C. 6LoWPAN Gateway System Components

There are many components within gateway that are important for the end-to-end system to work properly. This paper focuses on the packet management within the Gateway. Two main components in the gateway system, which are the focus of this paper that are used so that the packets are properly translated and forwarded are:

- **Node Management** - consists of Node Discovery (ND), Periodical Logger, Mapping Table and Predefined IPv6 Prefix and Address Translation.
- **Packet Handling and Translation** - consists of IPv6 Packet Handler, IPv6 - 6LoWPAN Packet Transformation and Predefined Remote Station Address.

The Node Discovery is a service that discovers the list of node as well as informing the nodes in 6LoWPAN network about their gateway. Both the gateway and the WSN nodes must have the Node Discovery module. The Node Discovery can be active or passive. For the active Node Discovery, the gateway will periodically broadcast Gateway Advertisement (GW_ADV) packet through IEEE802.15.4 interface to 6LoWPAN network. The nodes will response to this GW_ADV message with advertisement response (ADV_RESPONSE). Using

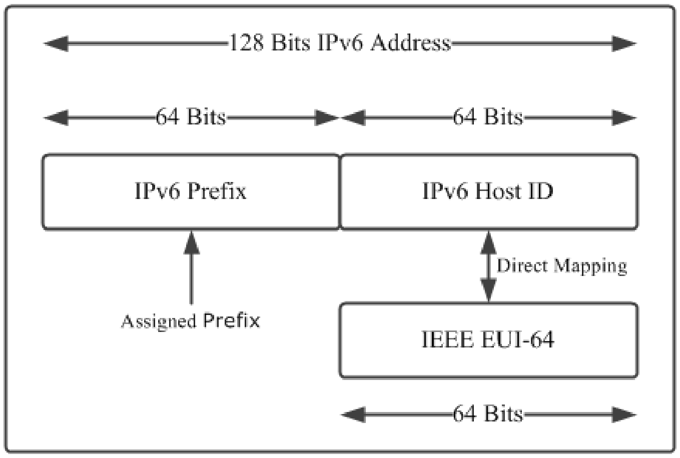


Fig. 6. IPv6 Address Assignment to 6LoWPAN Nodes

this option, the gateway can retrieve information of any sensor nodes available within 6LoWPAN network. Moreover, the nodes will also know their gateway that interfaces with the external IP network. The process of gateway sending GW_ADV message and node response with ADV_RESPONSE message is called Network Join Process. In addition, the MAC addresses of the 6LoWPAN nodes are retrieved from the ADV_RESPONSE message and stored in the Mapping Table. Thus, the Mapping Table for address translation will be generated from the network join process.

The translation is executed after the gateway receives the ADV_RESPONSE by adding a predefine 64 bit IPv6 prefix to a MAC address (EUI-64 bit) of a sensor node, which is retrieved from the MAC header of 6LoWPAN packet. Using this approach, the gateway manages the pseudo IPv6 address of the sensor node. Therefore, the gateway can ignore the process of sending out prefix advertisement to the 6LoWPAN network. This process provides some benefits.

- Message overhead would be reduced as prefix is not sent to the nodes
- Nodes would not process prefix configuration and hence power is not used unnecessarily
- Nodes does not have to allocate memory to configure the IPv6 address

The EUI-64 identifier of a 6LoWPAN device can be used as the interface identifier of the IPv6 address while the predefine IPv6 prefix is used as network identifier. Since the EUI-64 addresses are globally unique and appending it to IPv6 prefix to generate IPv6 addresses are globally unique as well. Figure 6 shows the address translation process and Table I illustrates the mapping table maintained by the gateway after the translation process.

D. Operation and Communication of 6LoWPAN Gateway

To give clear understanding on the practical use of this system using pull based mechanism, Figure 7 provide detail network time diagram.

TABLE I
EXAMPLE OF SENSOR NODE MAPPING TABLE

IPv6 Address	EUI-64 MAC Address
2001:2B8:F2:1:**7E23:1200:20:1200**	**7E:23:12:00:00:20:12:00**
2001:2B8:F2:1:**7D10:400:206:1501**	**7D:10:04:00:02:06:15:01**

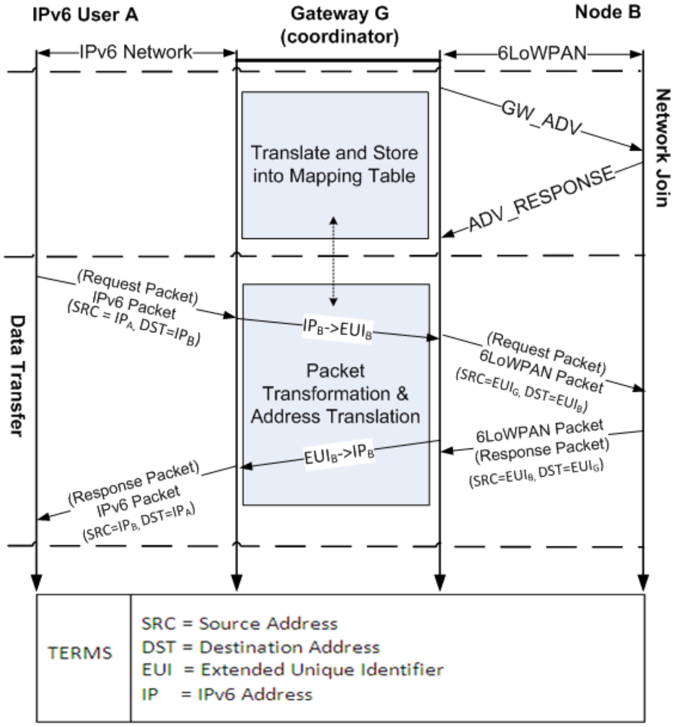

IPv6 User A
Gateway G (coordinator)
Node B
IPv6 Network
6LoWPAN
Translate and Store into Mapping Table
GW_ADV
ADV_RESPONSE
Network Join
Data Transfer
(Request Packet) IPv6 Packet (SRC = IP$_A$, DST=IP$_B$)
IP$_B$->EUI$_B$
Packet Transformation & Address Translation
(Request Packet) 6LoWPAN Packet (SRC=EUI$_G$, DST=EUI$_B$)
6LoWPAN Packet (Response Packet) (SRC=EUI$_B$, DST=EUI$_G$)
EUI$_B$->IP$_B$
(Response Packet) IPv6 Packet (SRC=IP$_B$, DST=IP$_A$)
TERMS
SRC = Source Address
DST = Destination Address
EUI = Extended Unique Identifier
IP = IPv6 Address

Fig. 7. Pull-Based Communications

Nodes in the wireless sensor network are first need to be registered in the gateway by following network join process explained earlier. This process is executed only at the beginning of network setup and periodically thereafter. This is similar to standard IPv6 neighbor discovery (ND) [18], wherein the advertisements from routers are sent periodically. The periodic time is set at larger intervals to reduce message overhead hence reduces the power consumption for processing the messages. Following are the steps taken in the network join process as given in Figure 7:

- Gateway G conduct node discovery by issuing a GW_ADVERTISEMENT message to the 6LoWPAN network.
- Node B, upon receiving this message, responds with ADV_RESPONSE message indicating that it will join the network.
- Gateway G will update the table with the information of the nodes responded

New nodes that join the network can update their presence using network join message, NET_JOIN message. Nodes can send this message if they did not receive any gateway discovery message from the gateway after a predefined time.

TABLE II
DETAIL INFORMATION IN ADDRESS INFORMATION TABLE

FIELD	LENGTH	DESCRIPTION
ID	1 byte	The requesting packet sequence number
Source Address	16 bytes	IPv6 address of the user (client)
Destination (Sensor Node) MAC Address	8 bytes	The MAC address (EUI-64 bit) of the sensor node. The address is derived by removing the IPv6 prefix from the sensors IPv6 address
Port	2 bytes	Port number allocated (from 61616 - 61630)
Status	1 byte	0: Packet has been forwarded to 6LoWPAN node. 1: Packet has been forwarded to IPv6 client. 2: Pending because the destination address is the as previous packet, which has not been received the response from the node.

The communications for both push based and pull based schemes are maintained through the use of a gateway. Different port numbers are used to differentiate the sensor's traffic for both the schemes. RFC 4944 [19] defines a well-known port range (61616-61631) for UDP packet in 6LoWPAN. In this implementation, the ports used are as follows:

- Port 61616 is used by the gateway to send data to the sensor nodes in pull based mechanism.
- Port 61617 is used by the gateway to receive data from sensor nodes in pull based mechanism.
- Port 61630 is used by the nodes to receive the request from the external node through the gateway and response using the same port.
- Port 61631 is used at the gateway to receive data from sensor nodes in push based method.

For both the communication mechanisms, Gateway maintains an Address Information Table as given in Table II. The gateway can differentiate traffic to the specific nodes that uses the ports defined and traffic from other applications. This is by referring to the table that has been created to store all the nodes that would use these ports. If there are other applications that uses different ports, the system would then operate as defined in the standard.

One of the examples of polling wireless sensor data is using one to one Communication.

This communication scenario occurs whenever different IPv6 clients request data from different sensor nodes. As an example, as shown in Figure 8, 2 IPv6 clients and 2 sensor nodes connected through a Gateway are used. Each IPv6 client requests data from different sensor nodes in 6LoWPAN network.

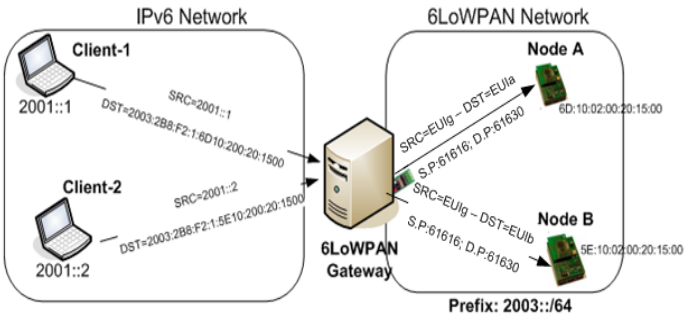


Fig. 8. IPv6 Client requesting data from sensor node

TABLE III
ADDRESS INFORMATION TABLE UPON RECEIVING REQUESTS FROM AN IPV6 CLIENT

ID	IPV6 SOURCE ADDRESS	DESTINATION MAC ADDRESS	PORT	STATUS
1	IP_1 (2001::1)	6D:10:02:00:20:15:00	61616	0
2	IP_2 (2001::2)	5E:10:02:00:20:15:00	61616	0

Based on Figure 8, upon receiving the IPv6 packet requests from an IPv6 client, the gateway will execute Forwarding Process for each packet:

i. Gateway updates the entry for the Address Information Table (Table III) by storing the Destination MAC Address field in the table, which is derived by removing the IPv6 prefix (2003:2b8:f2:1) used for sensor nodes. The gateway do not keep IPv6 destination address (sensor's IPv6 address) since the address can be generated by adding the prefix (e.g., 2003:2b8:f3:1) with EUI-64 address.
ii. The gateway checks the destination address (EUI-64 address). If there is an earlier request for data from the same address (status = 0) then the new request is queued by setting the status to 2.
iii. Once the packet is allocated with a source port, the gateway proceed to transform/convert the IPv6 packets to a 6LoWPAN packet:
    a. The gateway uses port number 61616 as source port. Port number 61630 is used for destination port at sensor node.
    b. Use the derived EUI-64 bit MAC address as destination address.
iv. The gateway forwards 6LoWPAN packet to 6LoWPAN network.

While processing any request packets, the gateway is ready for the reply from the sensor node. The Response Process for each response/reply packet from a sensor node is as follows:

i. The reply packet from sensor node will be sent to the port number 61617 of the gateway (Figure 9).
ii. The gateway will wait the reply for a certain amount of time (e.g., 1000 ms); if the gateway does not receive any reply, a second request message would be sent. If the gateway still did not receive any reply after that, it will send the *Time-Out Message* to the IPv6 client.

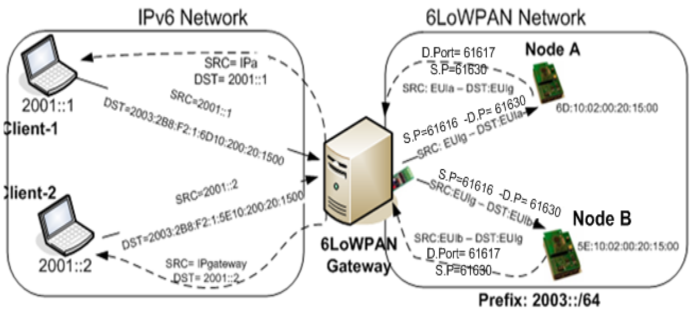


Fig. 9. Communication after receiving response from Sensor Node

TABLE IV
ADDRESS INFORMATION TABLE AFTER SENDING THE PACKET BACK TO IPV6 CLIENT

ID	IPV6 SOURCE ADDRESS	DESTINATION MAC ADDRESS	PORT	STATUS
1	IP_1 (2001::1)	6D:10:02:00:20:15:00	61617	1
2	IP_2 (2001::2)	5E:10:02:00:20:15:00	61617	1

iii. After the gateway receives a reply from the sensor node, it checks the Address Information Table, and matches the EUI-64 source address of the reply packet in order to retrieve IPv6 address of the client (IPv6 Source Address). The IPv6 source address will be used as the destination address to route back the packet to IPv6 client.

iv. Next, the 6LoWPAN packet is converted to IPv6 packet and route it back to IPv6 client.

v. The Status Field in the table is set 1, meaning the reply packet from sensor node already forwarded to IPv6 client (Table IV) and the entry in the Address Information Table will be deleted.

III. IMPLEMENTATION AND TESTING

A testbed was created to validate the gateway architecture and to measure the end-to-end performance as shown in Figure 10.

The setup consists of nano router and sensor nodes developed by Sensinode Inc. [21] as our hardware platform. Gateway is a laptop computer with Linux OS and has three interfaces; a nano router for the wireless sensor network, WiFi network interface and Ethernet network interface that connects to the IPv6 network. Nano router is a USB device that is attached to one of the available USB port in the gateway. Packet Handler module explained earlier is configured and executed on the gateway. The sensor nodes are installed with the free real-time operating system (FreeRTOS) with the NanoStack software module, which consists of 6LoWPAN stack with added features. Each of the sensor node has 2 AA batteries. The modules were developed using c programming language. The communications for both push based and pull based schemes are maintained through the use of a gateway. The sensor nodes that were deployed provide readings for temperature and light intensity measurements.

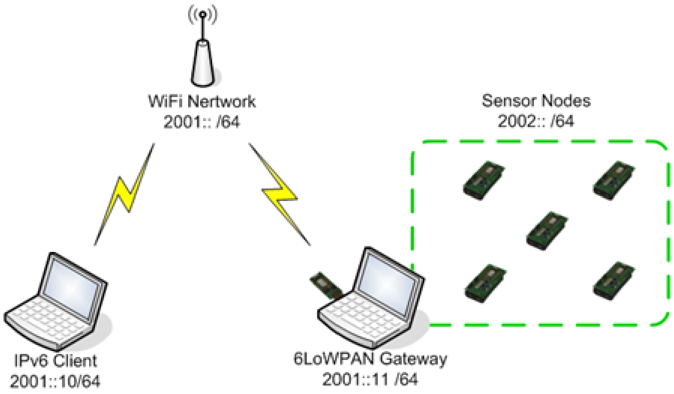


Fig. 10. Testbed for the System

TABLE V
PERFORMANCE MEASUREMENT PROPERTIES

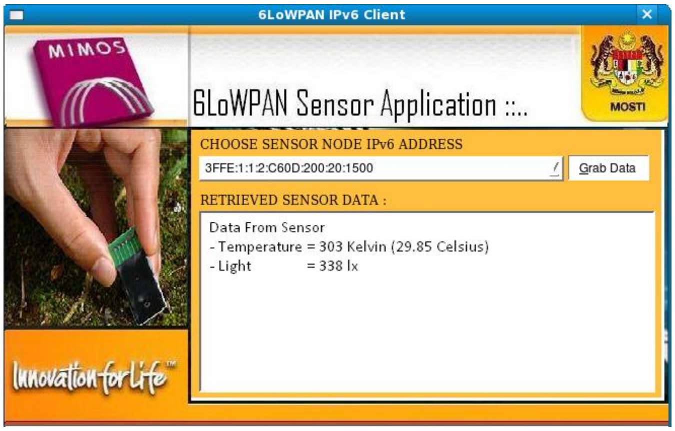
Properties	Details
Network Size	4-8 nodes for 1 hop away. 2x2, 2x4 and 2x6 for 2 hops
Distance	3 meters for each hop
Data Sampling intervals	20 seconds
Duration	120 samples (1 hour)
Message size	4, 8, 16, 37 bytes
Measurements	Transmission Success Rate and Latency
Method	Start with 1 node and gradually increase the nodes while sending data simultaneously

A client laptop was also used to retrieve sensor data to verify the bidirectional communication. To validate the performance, tests with different settings were conducted with different data sizes. Furthermore, to test the bidirectional communication, a ping message was sent from the gateway and using the reply, the latency was calculated. Table V provides the properties for the tests.

In 2 hops network environments, the end sensor nodes are configured to forward data through a particular relay node. In the experiments conducted, the sensor nodes are divided equally among the relay nodes. In 2x2 network setup, 1 sensor node forwards data through 1 relay node, in 2x4 network setup, 2 sensor nodes forward data through 1 relay node and in 2x6 network setup, 3 sensor nodes forward data through 1 relay node.

Wireless sensor nodes used in the experiments are configured with the following features:

- The sensor nodes are static (no mobility)
- The nodes are configured without any sleeping schedule hence the nodes will always be active to send and receive data
- Nodes are configured to forward the packets to the gate-



6LoWPAN IPv6 Client

6LoWPAN Sensor Application ::..

CHOOSE SENSOR NODE IPv6 ADDRESS

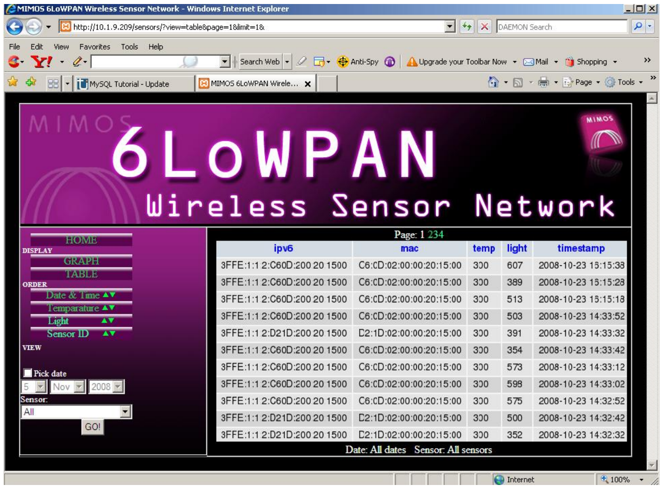
3FFE:1:1:2:C60D:200:20:1500

Grab Data

RETRIEVED SENSOR DATA :

Data From Sensor
- Temperature = 303 Kelvin (29.85 Celsius)
- Light = 338 lx

Fig. 11. IPv6 client application to read data directly from sensors. ©2009 MIMOS Bhd. All Rights Reserved



MIMOS 6LoWPAN Wireless Sensor Network - Windows Internet Explorer

http://10.1.9.209/sensors/?view=table&page=1&limit=1&

6LoWPAN
Wireless Sensor Network

HOME
DISPLAY
GRAPH
TABLE
ORDER
Date & Time ▲▼
Temparature ▲▼
Light ▲▼
Sensor ID ▲▼
VIEW

□ Pick date
5 Nov 2008
Sensor:
All
GO!

Page: 1 234

ipv6	mac	temp	light	timestamp
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	607	2008-10-23 15:15:38
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	389	2008-10-23 15:15:28
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	513	2008-10-23 15:15:18
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	503	2008-10-23 14:33:52
3FFE:1:1 2:D21D:200 20 1500	D2:1D:02:00:00:20:15:00	300	391	2008-10-23 14:33:32
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	354	2008-10-23 14:33:42
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	573	2008-10-23 14:33:12
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	598	2008-10-23 14:33:02
3FFE:1:1 2:C60D:200 20 1500	C6:CD:02:00:00:20:15:00	300	575	2008-10-23 14:32:52
3FFE:1:1 2:D21D:200 20 1500	D2:1D:02:00:00:20:15:00	300	500	2008-10-23 14:32:42
3FFE:1:1 2:D21D:200 20 1500	D2:1D:02:00:00:20:15:00	300	352	2008-10-23 14:32:32

Date: All dates Sensor: All sensors

Fig. 12. Display sensor information using web browser. ©2009 MIMOS Bhd. All Rights Reserved

way through a relay node in a 2 hops static deployments

IV. System Performance Evaluation

As described earlier, the request from a client will be forwarded by the gateway using a simple client as shown in Figure 11 [4]. All the sensor nodes' IPv6 addresses are listed in the client and when a particular IPv6 address is selected, a request is forwarded to the gateway, which will then do the necessary actions. The temperature and light reading from the sensor will then displayed on the client. This shows the success of bidirectional communication (Pull based mechanism). In the push based mechanism, the data is periodically sent to a web server and the data is displayed using a web browser as shown in Figure 12.

End-to-end latency usually measured using the ping command by getting the round trip time (RTT). The one way latency is half of the RTT value. There are few components that contributes to the end-to-end latency as given below.

- Processing of the packets - This latency is due to the processing power available at both end nodes. Request packet sent from the application layer has to move to the physical layer so with low processing at the node increases the latency but this is usually minimal.
- Network processing - In a typical network environment, the packets traverse through many routers and processing of packets at the router further increases the latency. The queueing delay is under this category. This happens when a gateway receives multiple packets from different sources heading towards the same destination. This problem is tackled using the Packet Management Module on the gateway.
- Network condition - The network condition is usually unpredictable hence if the network is congested, the packets that travel will get delayed and further increases the latency. Latency value is even more if the packet is sent in a wireless environment. In wireless multi hop environment, inefficient quality of service also effects the latency.

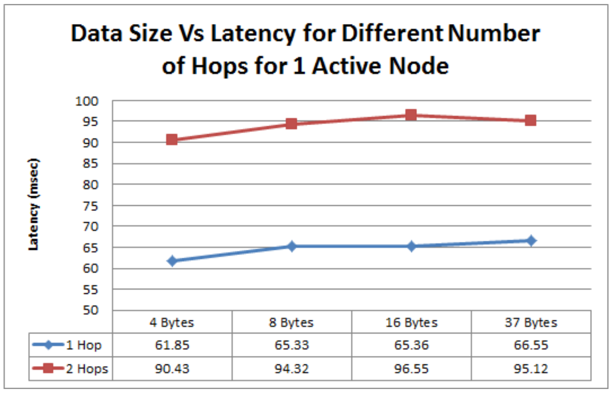
The end-to-end latency when only 1 sensor node is active is measured and average latency is 64.7 milliseconds for 1 hop and 94.1 milliseconds for 2 hops. This average latency is comparable with average latency claimed in the white paper by IPSO-Alliance [22], which is about 125 milliseconds. Total latency is calculated based on the processing latency of packet at the node, processing latency at the network gateway or router and latency due to network condition.

The latency for various data sizes for only 1 node active are given in Figure 13. It can be observed that the increase of data size does not effect much on the latency. This is because the packets are not fragmented and data is sent in one packet. However, the latency increases with the increase of number of hops. This is because of the effect of network condition that is explained earlier. In 2 hop network setup, the packets have to send to the relay node before being sent to the gateway. Processing of packets at the relay node further adds the latency. Figure 14 and Figure 15 show the average for 1 hop and 2 hops. These results are used as a base for the other experiments.

In Figure 16 and Figure 17, it can be observed that as the nodes increased one by one, the latency is also increases. This is because the nodes started sending data every 15 seconds after associating with the gateway. The increase in latency is due to the network condition component explained earlier, which is the increase of active nodes in wireless network increases the latency.

As for the packet delivery rate, 100% success rate obtained for nodes 1 hop away from the gateway for all the scenarios. However, the percentage dropped with the increase of hops and nodes as shown in Figure 18. This is due to the condition of the WSN network and the relay node are not configured with proper packet handling.

To further validate that the system developed is better in terms of packet delivery rate, two experiments, which are without the data management module were executed. The graphs for packet delivery rate for 1 hop and 2 hops without Packet Management Module proposed are given Figure 19 and



	4 Bytes	8 Bytes	16 Bytes	37 Bytes
1 Hop	61.85	65.33	65.36	66.55
2 Hops	90.43	94.32	96.55	95.12

Fig. 13. Data Size Vs Latency for Different Hops for 1 Active Node

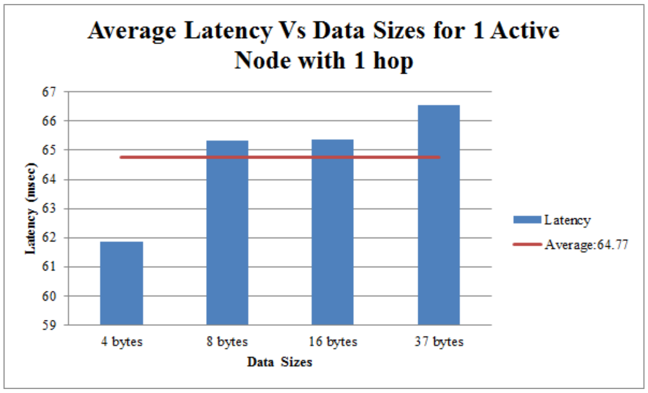
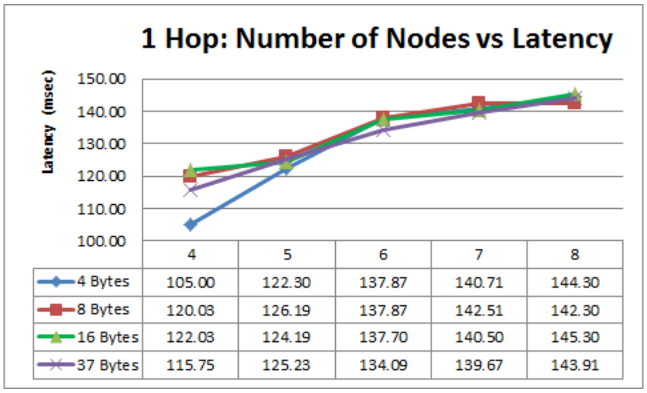
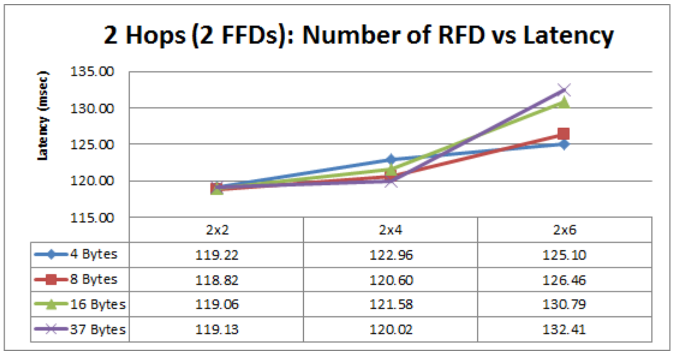


Fig. 14. Average Latency Vs Data Sizes for 1 Active Node with 1 hop



	4	5	6	7	8
4 Bytes	105.00	122.30	137.87	140.71	144.30
8 Bytes	120.03	126.19	137.87	142.51	142.30
16 Bytes	122.03	124.19	137.70	140.50	145.30
37 Bytes	115.75	125.23	134.09	139.67	143.91

Fig. 16. Number of Nodes Vs Latency in 1 Hop Network Environment



	2x2	2x4	2x6
4 Bytes	119.22	122.96	125.10
8 Bytes	118.82	120.60	126.46
16 Bytes	119.06	121.58	130.79
37 Bytes	119.13	120.02	132.41

Fig. 17. Number of Nodes Vs Latency in 2 Hops Network Environment

Figure 20.

It can be observed from Figure 19 that there is significant drop of packet delivery rate from 99% for 4 actives nodes and 4 bytes of data to 89% for 8 active nodes and 37 bytes of data. System with the data management module gives 100% packet delivery rate. This is because the gateway without the data management module receives packets from different nodes at the same port and could not handle the packets properly.

Since the same relay node was used, the packet delivery rate dropped with the similar margin as in 1 hop compared to the use of data management module. This is because the relay node is not configured with data management. The drop further increases because no data management at the gateway. The packet drop for 1 hop ranges from 1% to 13% whereas for 2 hops, the packet drop ranges from 2% and 4%.

This shows the importance of packet management at the gateway. With the proper address and data management, packet delivery rate for wireless sensor node can be improved.

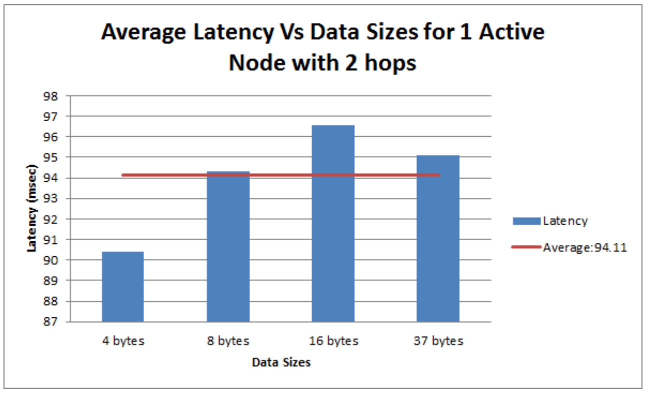
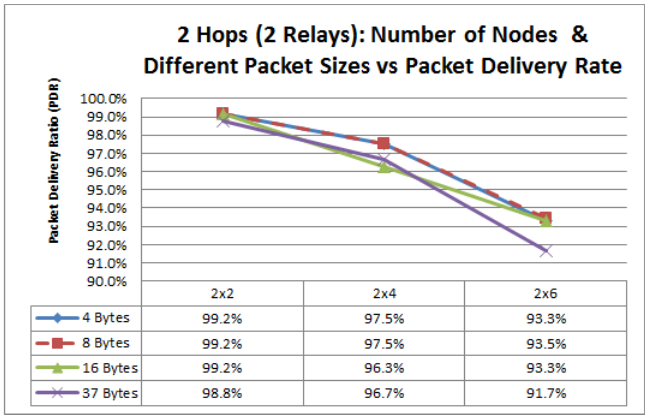
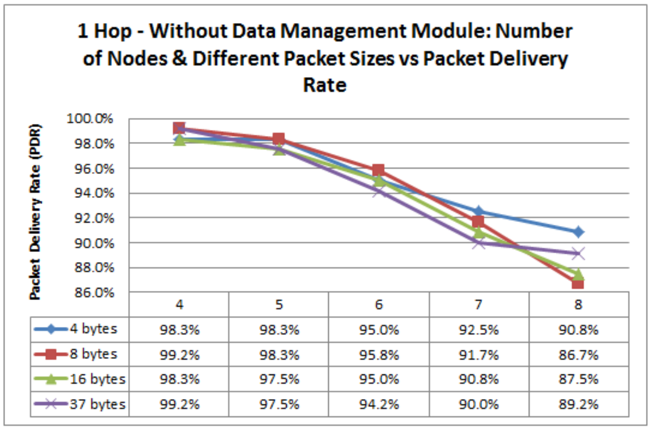


Fig. 15. Average Latency for 1 node in 2 hops with difference data sizes



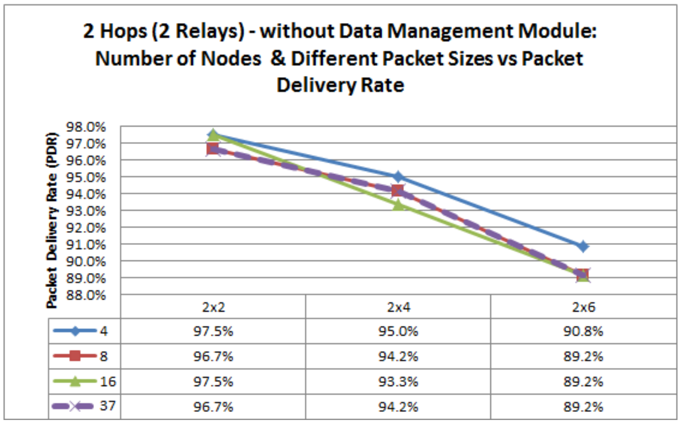
	2x2	2x4	2x6
4 Bytes	99.2%	97.5%	93.3%
8 Bytes	99.2%	97.5%	93.5%
16 Bytes	99.2%	96.3%	93.3%
37 Bytes	98.8%	96.7%	91.7%

Fig. 18. Number of Nodes Vs Packet Delivery Rate in 2 Hops Network Environment



	4	5	6	7	8
4 bytes	98.3%	98.3%	95.0%	92.5%	90.8%
8 bytes	99.2%	98.3%	95.8%	91.7%	86.7%
16 bytes	98.3%	97.5%	95.0%	90.8%	87.5%
37 bytes	99.2%	97.5%	94.2%	90.0%	89.2%

Fig. 19. Number of Nodes Vs Packet Delivery Rate in 1 Hop Network Environment without Data Management Module



	2x2	2x4	2x6
4	97.5%	95.0%	90.8%
8	96.7%	94.2%	89.2%
16	97.5%	93.3%	89.2%
37	96.7%	94.2%	89.2%

Fig. 20. Number of Nodes Vs Packet Delivery Rate in 2 Hops Network Environment without Data Management Module

V. WSN Gateway Related Work

There are several gateway architectures that were proposed for various implementation scenarios. Initially WSN was deployed in isolated network because of the constraint of the devices and technologies. With the progress of technology, data from WSN nodes was collected by the collector and send to a centralized server using GSM network or long range radio. With the introduction of web communication, data can be displayed in the web server but the data is still collected by the collector. Now, this devices have better processing capability and the need for the nodes to be connected to external network are more prevalent so IP connectivity has been suggested. This is possible with the use of gateway for communication to external network.

The systems are grouped based on the trend specified and each system is described by their research contributions and implementations. Three type of connectivity methods are discussed with the emphasis given to the architecture used to implement and method of managing the packets at the gateway. Some systems are deployed using the proprietary protocol such as ZigBee while others using open source such as TinyOS, Contiki and Nanostack.

A. Gateway to server type of connectivity

In this method, data from sensor nodes is sent to a gateway, which then forwards to a server. Gateway may have different type of connectivity to the server such as GSM, GPRS or others. In the system developed by Steenkamp et al. at Cape Penisular University of Technology [5], WSN gateway was developed using TinyOS with AT91RM9200 ARM evaluation kit from Atmel. This gateway enables users to remotely retrieve data from WSN network using GSM network.

A system specifically designed to gather information from the forest was proposed by Wenbin et al. [6]. In this system, gateway is connected to an external server using GPRS module. The gateway collects sensor data and converts it into Comma-Separated Value (CSV) format. After that the CSV file is sent to the server using FTP via GPRS module. Communication between the gateway and the FTP server is established using TCP/IP protocol that was built-in in the Debian Linux for embedded devices. Another system developed using GPRS module was implemented by Tolle et al. [7]. Data from the sensor nodes was collected over Mica2 node attached to RS232 serial, stored in a local database, and then transmitted over a GPRS cellular modem to an off-site database. They implemented the system to capture the microclimate surrounding a coastal redwood tree.

A different approach was introduced by Becher et al. [8] to send health information to a personal computer. In this approach, person's health data such as ECG, pulse rate and body weight are sent to a gateway, which then forwards to a personal computer using Bluetooth technology. ZigBee communication was used between the WSN nodes and the gateway.

The systems and architectures given, uses a point-to-point communication between the gateway and the server using GPRS, Blueetooth, long range wireless or Satellite. The drawbacks with these system are:

- There is no data management at the gateway. Data is collected at the gateway, saved in a file and send to the other end. In some cases, data is forwarded as it arrived at the gateway. These systems are not suitable for critical applications because all the systems have a single point of failure. There is also no mechanism to determine if the data was successfully sent to its destination.
- It has no IP connectivity for the nodes as such end-to-end communication could not be performed. The data from the sensor is send to the gateway, which then forwards it to a data storage server. Information about the node's ID would be added in the data field and this cost more overhead. Besides that, data would be sent to a single end point like the system that uses Bluetooth [8] and not routable in the Internet. IP address given to each node would enable the nodes to be reachable from anywhere as.

B. Communication using Web Services

Systems developed using this method uses web services to publish the collected data. The web service may be running in a separate server or part of gateway. In a system proposed by Qiu and Choi [9], the information from the sensor nodes are displayed using web server. The approach taken was to setup a web server in the gateway itself using embedded Common Gateway Interface (CGI) technology. Users can check the data from ZigBee sensor network through the web-sensor gateway. Users can get data from a particular sensor by sending a request through the web server at the gateway. The gateway, after receiving the information using the ZigBee protocol, displays the information on the web server for the client to view.

Fan Ye Dun et al. [10] presents a gateway, which connects WSN with external network. In this gateway that uses TinyOS, data is gathered at the gateway and stored in the local storage using embedded database, SQLite3. The information in the database is displayed using a web service so that any external user can access and view the data using existing TCP/IP protocol. Overall architecture presented is similar to system [9] discussed earlier. This system was used for environmental monitoring. The limitation with this approach is similar to the earlier system, which inhibits the end-to-end communication that is important in some applications. Maybe the system is only suitable for environmental monitoring and not for other use cases. Another similar system that displayed data using web services was proposed by Dan et al. [11]. The data are stored in Extensible Markup Language (XML) files according to information types. Web service interface within the gateway encapsulates XML format data in Simple Object Access Packet (SOAP) packet and transmitted to web browser through HTTP protocol. Similar concept was also used in a system developed by Jin et al. [12] for home and building automation and by Malatras et al. [13] for facility management.

Some of the drawbacks with the systems discussed in this category are:

- Nodes cannot be reachable directly from the external network. This limits the goal of providing direct communication and limits the growth of WSN in other aspects such as mobility, etc. The gateway requires extra resources as it also provides web services and data storage. In some systems, IPv4 address was assigned based on the availability. This further inhibits the growth of the network.
- There were no proper data management at the gateway. It is not necessary for this group because most of the time, it is communication between the sensor nodes and the gateway.

C. End-to-end connectivity

In this system, WSN nodes somehow are able to connect to external network using few methods. Zimmermann et al. [14] developed a system using a combination of DNS reverse lookup and address translation method to extend WSN node to external network. In this system, the sensor nodes are configured with IPv6 link local address. Each of the nodes is mapped to a global IPv6 address using the 1 on 1 Network Address Translation (NAT) mapping. Whenever a node wants to communicate with an external device, it is assumed that the node knows the domain name of the external device and sends the query for IPv6 address. The gateway forwards the query while maintaining the requester's information in its database. The Domain Name System (DNS) Application level gateway intercepts the query and replaces the domain name with global IPv6 address of the external device. The global address is mapped to a newly generated link local address using the 1 on 1 NAT mapping at the gateway. The limitations with this system are:

- If the DNS query is not intercepted for some reasons and the DNS server is heavily used, IPv6 address cannot be returned to the sensor node. This will fail the communication between the sensor node and the external device.
- There is no management of the packets at the gateway besides the 1 to 1 mapping. Using link local address adds extra overhead on the node. This can be reduced by reusing the MAC address already available in the header.
- This approach also uses extra overhead which consists of messages being exchanged to retrieve the external device's IPv6 address and this contributes to the increase of transmission latency.
- Both the sensor nodes and the external nodes have link local and global address, which is translated at the gateway using 1 on 1 NAT. The translation of the header increases processing of the packet and it is unnecessary.

An IP address translation mechanism was proposed by Choi et al. [15]. It is assumed that the gateway has records of all the external devices' IPv6 addresses. WSN node request the destination IPv6 address from the gateway by providing the link local address of the external device. Once the node receives the information, the node will then send the packet using EUI64 MAC address of the destination node. The gateway again change the MAC destination address to link local address. Even though the objective for this approach was to provide end-to-end communication, the approach taken was not practical.

- It is not practical for internal node to request address of the external device based on the link local address. In this implementation the gateway has to store all external devices' addresses, which is impossible. This is practical if the node sends data to a known address such as a server.
- Extra overhead and redundant message exchanges between the node and gateway. The node queries the gateway for destination ID by providing the destination link local address address. The destination node ID can actually be retrieved from the link local address used in the query. Furthermore, link local address is not routable in Internet thus it restricts the implementation to a particular local area network.
- There is no method mentioned on the management of data at the gateway. This would be a problem as in some

scenario, when nodes continuously and simultaneously transmit data to the gateway and without a proper management mechanism, packet loss will be high.

Since IPv6 is not fully deployed, Chang et al. [16] proposed and implemented a system using 6LoWPAN in IPv4 network. They propose that both public and private IPv4 address be used for the nodes in WSN. Connectivity from gateway to external network could be using Network Address Translator-Protocol Translator (NAT-PT), tunneling service such as ISATAP, Teredo, 6to4 and others.

ZigBee has been widely used in WSN and changing the protocol stack to support IPv6 is not practical hence Chia et al. [17] proposed an architecture using SIP protocol to interconnect ZigBee network with the external network. With this session layer approach, both ZigBee and 6LoWPAN WSN would be supported. For ZigBee nodes, the ZigBee Apps information is translated into SIP while SIP has to be supported in 6LoWPAN node. This extra layer service creates more overhead for 6LoWPAN nodes. End-to-end communication is not supported with this method and the architecture does not provide data management at the gateway.

There was various methods proposed to connect WSN to the Internet but none of it described in detail the method of end-to-end connectivity and does not provide data management at the gateway. Both the end-to-end connectivity and the data management are important features to be incorporated in WSN to ensure that data will be communicated effectively like any other Internet devices.

VI. Conclusion

This paper proposed a gateway system to interconnect wireless sensor network with external network using 6LoWPAN protocol. The gateway provides a mechanism for the end clients to directly communicate with the sensor node, which was assigned with IPv6 address. Besides that, the gateway forwards the periodical data to a web server.

The system is validated with the successful transmission of sensor data, which was displayed using a client and web server. Further tests were conducted to validate the latency and the transmission success rate. The latency for 1 hop with various number of nodes ranges between 60 to 145 milliseconds while the transmission success rate is 100 % for 1 hop. The success rate dropped with the increase of number of hop, which could be because of the relay node (FFD) not forwarding the packets appropriately. Nevertheless, the results are in accordance with the other prior art. It is expected that further increase in the number of hops would reduce the packet transmission success rate.

As for future work, the proposed solution can be further tested in other environments by setting different transmission intervals, less interferences, etc. It is also important for the transmission to be extended with more than 2 hops with minimal packet drops. The performance can also be evaluated with the implementation of other components such as security, routing, dynamic topology and mobility with multi-hop scenarios. Besides that, the gateway can be extended to be used as IoT gateway that will provide seamless connectivity to various standards and devices.

References

[1] Gopinath Rao. S, Zeldi Suryady, Usman Sarwar, Mazlan Abbas, and Sureswaran Ramadass, “IPv6 Wireless Sensor Network Gateway Design and End-to-End Performance Analysis”, SENSORCOMM 2012, The Sixth International Conference on Sensor Technologies and Applications, held in Rome, Italy, pp. 67-72, August 19-24, 2012.

[2] Institute of Electrical and Electronics Engineers (IEEE), “IEEE 802.15.4.” http://standards.ieee.org/about/get/802/802.15.html

[3] IPv6 over Low Power Personal Area Network (6LoWPAN) IETF Working Group. Retrieved: July, 2012. http://datatracker.ietf.org/wg/6lowpan/

[4] G. R. Sinniah, Z. Suryady, U. Sarwar, and M. Abbas, “A Gateway Solution for IPv6 Wireless Sensor Network”, Ultra Modern Telecommunication & Workshops, St. Petersburg, Russia, pp. 1-6, October 2009.

[5] Steenkamp, L. and Kaplan, S. and Wilkinson, R.H., “Wireless sensor network gateway”, The 9th IEEE AFRICON 2009, pp. 1-6, September 2009.

[6] Li Wenbin, Cui Dongxu, and Zhang Junguo, “Design and Implementation of Wireless Sensor Network Gateway Faced to Forest Information Monitor”, 2010 International Conference on Intelligent System Design and Engineering Application (ISDEA), pp. 524-526, October 2010.

[7] Gilman Tolle, Joseph Polastre, Robert Szewczyk, David Culler, Neil Turner, Kevin Tu, Stephen Burgess, Todd Dawson, Phil Buonadonna, David Gay, and Wei Hong, “A macroscope in the redwoods”, Proceedings of the 3rd international conference on Embedded networked sensor systems, SenSys ’05, San Diego, USA, pp. 51-63 2005.

[8] K. Becher, C.P. Figueiredo, C. Mu andhle, R. Ruff, P.M. Mendes, and K. Hoffmann, “Design and realization of a wireless sensor gateway for health monitoring”, 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Buenos Aires, Argentina, pp. 374-377 August 31 - September 4 2010.

[9] Peng Qiu, Ung Heo, and Jaeho Choi, “The web-sensor gateway architecture for ZigBee”, IEEE 13th International Symposium on Consumer Electronics, ISCE ’09, Kyoto, Japan, pp. 661-664, May 25-28 2009.

[10] Ye Dun-fan, Min Liang-liang, and Wang Wei, “Design and Implementation of Wireless Sensor Network Gateway Based on Environmental Monitoring”, International Conference on Environmental Science and Information Application Technology, ESIAT 2009, Wuhan, China, pp.289-292, 4-5 July 2009.

[11] Dan Hu, Shi-Ning Li, and Zhi-Gang Li, “Design and Implementation of Wireless Sensor Network Gateway Based on Web Services”, 4th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM ’08, Dalian, China, pp. 1-4, 12-14 October 2008.

[12] Jin Seok Oh, Jeong Il Choi, Hyun Seok Lee, and Jeong Seok Heo, “Web-based real-time sensor monitoring system using Smart Client”, International Forum on Strategic Technology, IFOST 2007, Ulaanbaatar, Mongolia, pp. 619-622, 3 - 6 October 2007.

[13] A. Malatras, A. Asgari, and T. Bauge, “Web Enabled Wireless Sensor Networks for Facilities Management”, IEEE Systems Journal, pp. 500-512, December 2008.

[14] A. Zimmermann, J. Sa Silva, J.B.M Sobral, and F. Boavida, “ 6GLAD: IPv6 Global to Link-layer Address Translation for 6LoWPAN Overhead Reducing”, 4th EURO-NGI Conference on Next Generation Internet Networks, NGI 2008, Krakow, Poland, pp. 209-214, 28-30 April 2008.

[15] Dae-In Choi, Jong-tak Park, Su-yoen Kim, and H.K. Kahng, “IPv6 global connectivity for 6LoWPAN using short ID”, 2011 International Conference on Information Networking, (ICOIN), Kuala Lumpur, Malaysia, pp. 384-387, 26-28 January 2011.

[16] Chang-Yeol Yum, Yong Sung Beun, Sunmoo Kang, Young Ro Lee, and Joo Seok Song, “Methods to use 6LoWPAN in IPv4 network”, The 9th International Conference on Advanced Communication Technology, (ICACT), Gangwon-Do, Republic of Korea, pp. 969-972, 12-14 February, 2007.

[17] Chia-Wen Lu, Shu-Cheng Li, and Q. Wu, “Interconnecting ZigBee and 6LoWPAN wireless sensor networks for smart grid applications”, The Fifth International Conference on Sensing Technology (ICST), Palmerston North, New Zealand, pp. 269-272, 28 November- 1 December 2011.

[18] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, RFC 4861: Neighbor Discovery for IP version 6 (IPv6), IETF Standard Document, September 2007.

[19] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, RFC 4944: Transmission of IPv6 Packets over IEEE 802.15.4 Networks, IETF Standard Document, September 2007.

[20] J. Hui and P. Thubert. RFC 6282: Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks, IETF Standard Document, September 2011.

[21] "Sensinode hardware and NanoStack Operating System", 2008. Retrieved: July, 2012. Available: http://www.sensinode.com/

[22] J. Abeill, M. Durvy, J. Hui, and S. Dawson-Haggerty, "Lightweight IPv6 Stacks for Smart Objects: the Experience of Three Independent and Interoperable Implementations", November 2008. Available at: http://www.ipsoalliance.org/white-papers